

GenoMetric Query Language (GMQL)

Quick Start

Genomic Computing Group

Dipartimento di Elettronica, Informazione e Bioingegneria
Politecnico di Milano

November 1, 2014

1 GMQL Working Modes

GMQL has two working modes that differ in processing capabilities.

1. **MAPREDUCE mode**

The power of GMQL is its ability to run complex, big data queries on huge input datasets in a cluster of machines. In the **MAPREDUCE** mode, GMQL uses the cloud computing Hadoop Mapreduce engine for processing and the Hadoop Distributed File System (HDFS) to distribute and maintain the data in a cluster. To run it, an Hadoop installation is required.

2. **LOCAL mode**

In the **LOCAL** mode the GMQL toolkit uses only the resources (RAM, CPU and storage) available on the computer that runs GMQL; it works on the local file system (LFS) for all file operations, without duplicating or distributing any sample data file. The **LOCAL** mode allows easy installation and testing of the toolkit, but it has limited performance, sufficient just for processing a limited data size.

2 Dependencies

In order to run, GMQL requires the user to be using Bash; this is the default option in almost all the modern GNU/Linux distributions.

In most of the GNU/Linux distributions, it is possible to install both Java 7 and Racket v.5.3 or later through the package manager. If this is not possible in the used distribution, please follow these instructions, by specifying the **full path** for the variables in every parameter:

JAVA JDK 7.

- You can download JDK 7 from:
`http://www.oracle.com/technetwork/java/javase/downloads/`
- Untar the java package into a specific folder, for example:
`/home/<your_user>/java7/`
- Then, add the following line to the PATH:
`export PATH=/home/<your_user>/java7/jdk1.7.0_25/bin:$PATH`

Racket v5.3 or later.

- You can download Racket from:
`http://racket-lang.org/download/`
- Then, add the following lines to the PATH:
`export RACKET_HOME=/path_to_your_racket_installation`
`export PATH=$PATH:$RACKET_HOME/bin/`

Hadoop (required only for MAPREDUCE mode).

- For Hadoop V 1.x see for example:
`http://hadoop.apache.org/docs/r1.2.1/single_node_setup.html`

3 Setting the Environment Variables - LOCAL mode

To use the default values for the environment variables, append the content of the `conf/GMQL-env.sh` file, located in the `conf/` folder, to the `~/.bashrc` file (or the `~/.bash_profile` file in older Linux versions), located in the home folder of each user who will use GMQL. Do not run `conf/GMQL-env.sh`

Note: Save any file change **only** in Linux, to avoid malfunctioning.

To change the default values of the environment variables, do as follows **both** in the `conf/GMQL-env.sh` file and the `~/.bashrc` file (or the `~/.bash_profile` file in older Linux versions):

- Set the directory path for java JDK (the default is: `/usr/lib/jvm/java-7-oracle/`). If Java has been installed as described in Section 2, add:

```
export JAVA_HOME=/home/<your_user>/java7/jdk<your_jdk_version>/
```

otherwise add:

```
export JAVA_HOME=/home/path_to_your_jdk_installation/
```

- Set the GMQL home, i.e. the location where all the local data of the GMQL repository, as well as the control and configuration data, are located (the default is: `/home/yourUserName/gmql_repository`):

```
export GMQL_HOME=~ /gmql_repository
```

- Set the environment variables for the Apache Pig installation, where Apache Pig V0.13.0 is automatically installed (the default is: `/userHomeFolder/pig`):

```
export PIG_HOME=~ /pig
export PIG_CONF_DIR=$PIG_HOME/conf
export PIG_CLASSPATH=$PIG_HOME/pig-0.13.0-h1.jar:
    $HADOOP_COMMON_LIB_NATIVE_DIR/lib/*:$PIG_CLASSPATH:
    $HADOOP_CONF_DIR:$PIG_CONF_DIR
```

To learn more about Pig environment variables, see the guide at:
<http://pig.apache.org/docs/r0.13.0/start.html>.

- Set the execution mode (the default is: `LOCAL`):

```
export GMQL_EXEC=LOCAL
```

4 Setting the Environment Variables - MAPREDUCE mode

- Set the directory path for java JDK (the default is: `/usr/lib/jvm/java-7-oracle/`). If Java has been installed as described in Section 2, add:

```
export JAVA_HOME=/home/<your_user>/java7/jdk<your_jdk_version>/
```

otherwise add:

```
export JAVA_HOME=/home/path_to_your_jdk_installation/
```

- Besides setting the execution mode to `MAPREDUCE` (i.e. `export GMQL_EXEC=MAPREDUCE`), set the following environment variables:

```
export HADOOP_HOME=/usr/local/hadoop
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_COMMON_HOME/lib/native
export HADOOP_CONF_DIR=$HADOOP_HOME/conf
```

- Set the GMQL home on the HDFS, where storing the sample data (the default is: /user/; note that here we mean exactly "user", not the user name):

```
export GMQL_DFS_HOME=/user/
```

- Finally, add the variables to the system PATH:

```
export PATH=$PATH:$JAVA_HOME/bin:$
HADOOP_HOME/bin:$PIG_HOME/bin:$GMQL_HOME/bin
```

5 Installing GMQL

1. Go to the folder .../GMQLPackage/
2. Run the following command:

```
source ~/.bashrc
(source ~/.bash_profile in older Linux versions)
```

3. Run the installer, i.e. ./install.sh and follow the instructions on the screen to install GMQL.
4. Run the following command to register your user to the GMQL repository:

```
repositoryManagerV1 RegisterUser
```

In case of multi-users of the GMQL system, each user must run this command from his/her environment.

The output of this command should look like:

```
INFO: Local Folders Creation ...
INFO: Folder /home/gmql_repository/data/username/indexes/ true
INFO: Folder /home/gmql_repository/data/username/datasets/ true
INFO: Folder /home/gmql_repository/data/username/metadata/ true
INFO: Folder /home/gmql_repository/data/username/schema/ true
INFO: Folder /home/gmql_repository/data/username/results/ true
INFO: Folder /home/gmql_repository/data/username/queries/ true
INFO: Folders are created
```